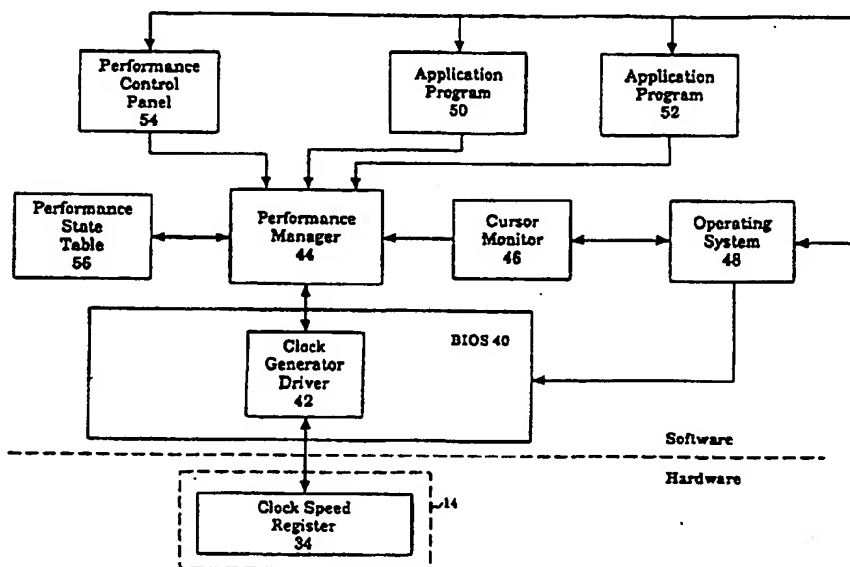




## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification <sup>6</sup> : <b>G06F 13/00</b>		<b>A1</b>	(11) International Publication Number: <b>WO 95/35540</b>
			(43) International Publication Date: 28 December 1995 (28.12.95)
(21) International Application Number: PCT/US95/07668 (22) International Filing Date: 14 June 1995 (14.06.95) (30) Priority Data: 08/261,457          17 June 1994 (17.06.94)          US (71) Applicant: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, CA 95052 (US). (72) Inventor: NORRIS, David; 19835 N.W. Nestucca Drive, Portland, OR 97229 (US). (74) Agents: SALTER, James, H. et al.; Blakely, Sokoloff, Taylor & Zafman, 7th floor, 12400 Wilshire Boulevard, Los Angeles, CA 90025-1026 (US).		(81) Designated States: AM, AT, AT (Utility model), AU, BB, BG, BR, BY, CA, CH, CN, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), EE, ES, FI, FI (Utility model), GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LT, LU, LV, MD, MG, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), TJ, TM, TT, UA, UG, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), ARIPO patent (KE, MW, SD, SZ, UG).  Published With international search report.	

(54) Title: DYNAMIC PROCESSOR PERFORMANCE AND POWER MANAGEMENT IN A COMPUTER SYSTEM



## (57) Abstract

A computer system (10) is disclosed comprising a clock generator circuit (14) having a clock speed register (34) and circuitry for generating a processor clock signal (32) at a frequency determined by the clock speed register (34), wherein the processor (12) executes a performance manager program (44) that writes the clock speed register (34) according to a performance state selected by an application program (50, 52). The application program (50, 52) selects the performance state to maximize performance during processor intensive functions and to maximize power conservation during interactive functions.

**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	GB	United Kingdom	MR	Mauritania
AU	Australia	GE	Georgia	MW	Malawi
BB	Barbados	GN	Guinea	NE	Niger
BE	Belgium	GR	Greece	NL	Netherlands
BF	Burkina Faso	HU	Hungary	NO	Norway
BG	Bulgaria	IE	Ireland	NZ	New Zealand
BJ	Benin	IT	Italy	PL	Poland
BR	Brazil	JP	Japan	PT	Portugal
BY	Belarus	KE	Kenya	RO	Romania
CA	Canada	KG	Kyrgystan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CG	Congo	KR	Republic of Korea	SE	Sweden
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LU	Luxembourg	TD	Chad
CS	Czechoslovakia	LV	Latvia	TG	Togo
CZ	Czech Republic	MC	Monaco	TJ	Tajikistan
DE	Germany	MD	Republic of Moldova	TT	Trinidad and Tobago
DK	Denmark	MG	Madagascar	UA	Ukraine
ES	Spain	ML	Mali	US	United States of America
FI	Finland	MN	Mongolia	UZ	Uzbekistan
FR	France			VN	Viet Nam
GA	Gabon				

**DYNAMIC PROCESSOR PERFORMANCE AND  
POWER MANAGEMENT IN A COMPUTER SYSTEM**  
**BACKGROUND OF THE INVENTION**

5   **1.   FIELD OF THE INVENTION:**

          The present invention pertains to the field of computer systems.  
More particularly, this invention relates to the dynamic management of  
processor performance and power consumption in a portable computer  
10   system.

**2.   BACKGROUND:**

          Recent portable computer systems such as laptop computers  
15   and palmtop computers typically contain a central processing unit and a  
variety of input/output devices such as a keyboard, a display device,  
and mass storage devices. Typically, such a portable computer system  
includes a battery subsystem that provides power to the computer  
system while alternating current power is not available. Such a battery  
20   subsystem is typically capable of supplying a limited amount of  
electrical current to the computer system before requiring a recharge or  
replacement.

          Such a prior portable computer system typically provides a  
25   variety of mechanisms for minimizing the electrical current consumption  
in the computer system. For example, such systems usually implement  
sleep modes and standby modes for the input/output devices that  
reduce electrical current consumption while the input/output devices  
are not in use. In addition such prior systems typically provide a clock  
30   speed control mechanism for the central processing unit.

          Typically, the central processing unit in such a system consumes  
less electrical current at lower processor clock speeds. Unfortunately,  
such lower power consumption is typically achieved at the expense of

performance by the central processing unit. The instruction throughput of such a central processing unit usually decreases in proportion to the decrease in the processor clock speed. As a consequence, the user of such a prior portable computer system must typically choose between a  
5 high performance, high power consumption state for the computer system or a low performance, low power consumption state for the computer system.

The consequences of slower processor clock speeds in such a  
10 computer system depends upon the function being performed by the executing program. The user typically does not perceive a significant reduction in performance during interactive functions while the processor is running at a slow clock speed. However, the user does perceive a significant degradation in performance during processor  
15 intensive functions at a slow clock speed. Such processor intensive functions are also referred to as processor bound functions.

For example, the user typically does not perceive a significant degradation in performance if the processor is set in a slow clock speed  
20 during interactive data entry and typing functions in a word processing application program. However, the user does perceive a significant degradation in performance during processor bound functions such as the reformatting of pages, the rendering data to a printer, the drawing complex diagrams or the reindexing data bases.

25 In such a prior system, the user may select a maximum performance level by setting the processor clock speed to a high frequency. In such a maximum performance mode, the user perceives maximum performance during processor bound functions.  
30 Unfortunately, the user typically experiences a significantly reduced battery life in such a high performance mode.

On the other hand, the user may select a maximum conservation mode by setting the processor clock speed to a low frequency. In such

a conservation mode, the user typically experiences extended battery life without a perceivable degradation in performance during interactive functions. Unfortunately, the user perceives significant degradation in performance during processor bound functions in such a maximum conservation mode.

In some prior portable computer systems, the user typically selects a desired processor clock speed using a function key sequence or a control panel graphical select function. Some prior portable computer systems allow the user to set the processor clock speed at any time without restarting the computer system. Nevertheless, such typical prior processor clock speed control mechanisms are too cumbersome to provide the user with any useful control over the balance between processor performance and system power consumption.

**SUMMARY AND OBJECTS OF THE INVENTION**

One object of the present invention is manage a balance between performance and power consumption in a computer system.

5

Another object of the present invention is to enable application programs in a computer system to select high performance modes of a processor during processor intensive functions.

10

Another object of the present invention is to enable application programs in a computer system to select power conservation modes of a processor during non processor intensive functions.

15

A further object of the present invention is enable a computer system user to select processor clock frequencies for a set of performance modes of a processor.

20

Another object of the present invention is to provide an accelerate button graphical interface that enables a user to increase the processor clock frequency during processor intensive functions.

25

These and other objects of the invention are provided by a computer system comprising a clock generator circuit having a clock speed register and circuitry for generating a processor clock signal at a frequency determined by the clock speed register. The computer system further comprises a processor coupled to receive the processor clock signal to synchronize the processor. The processor executes a performance manager program that writes the clock speed register according to a performance state selected by an application program executing on the computer system.

30

The processor writes the clock speed register with a value specified in a performance state table maintained by the performance manager program. The performance state table stores a set of clock

5

frequencies that correspond to a set of performance states for the processor.

5 Other objects, features and advantages of the present invention will be apparent from the accompanying drawings, and from the detailed description that follows below.

**BRIEF DESCRIPTION OF THE DRAWINGS**

The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings in which like  
5 references indicate similar elements, and in which:

**Figure 1** illustrates a computer system for one embodiment which comprises a processor, a clock generator circuit, a non-volatile memory, and a memory subsystem;  
10

**Figure 2** is a block diagram that illustrates the software architecture of the computer system for one embodiment which includes a performance manager, an operating system, and a set of application programs;  
15

**Figures 3a-3b** illustrate a control panel generated on a display device by a performance control panel which is an application program that enables a user of the computer system to select from a set of predetermined frequencies of the processor clock;  
20

**Figure 4** illustrates an information window for a word processing application program that provides an accelerate button;

**Figure 5** is a flow diagram that illustrates the handling of the processor performance states by an application program for one embodiment;  
25

**Figure 6** is a flow diagram that illustrates the functions of the performance manager for one embodiment.



**DETAILED DESCRIPTION**

**Figure 1** illustrates a computer system 10 for one embodiment. The computer system 10 comprises a processor 12, a clock generator circuit 14, a non-volatile memory 16, and a memory subsystem 18. The computer system 10 further comprises a set of input/output devices including a disk subsystem 20, a keyboard and mouse subsystem 22, and a display 24.

The processor 12 communicates with the non-volatile memory 16, the memory subsystem 18, the disk subsystem 20, the keyboard and mouse 22, and the display 24 via a system bus 28. The computer system 10 further comprises a battery subsystem 26. The battery subsystem 26 supplies an electrical direct current to the computer system 10 via a set of power lines of the system bus 28.

The non-volatile memory 16 provides a storage area for a basic input/output software (BIOS) for the computer system 10. The BIOS of the computer system 10 includes basic driver routines for controlling hardware functions of the disk subsystem 20, the keyboard and mouse 22, and the display 24. In addition, the BIOS of the computer system 10 includes a set of driver routines for accessing a clock speed register in the clock generator circuit 14.

The memory subsystem 18 provides storage areas for an operating system, application programs, and device driver programs and other associated data structures for the computer system 10. For one embodiment, the memory subsystem 18 comprises a dynamic random access (DRAM) memory subsystem.

The disk subsystem 20 provides large scale mass storage to the computer system 10 for programs and data. The keyboard and mouse subsystem 22 enables user interaction with the computer system 10. The display 24 provides display functions for the computer system 10.

The clock generator circuit 14 generates a processor clock 32 for the processor 12. The processor clock 32 is coupled to a clock input pin (CLK PIN) of the processor 12. The processor clock 32  
5 synchronizes the processor 12. The frequency of the processor clock 32 determines the rate at which the processor 12 fetches and executes instructions.

The processor 12 fetches an instruction stream from the memory  
10 subsystem 18 over the system bus 28. The processor 12 executes each instruction in the instruction stream. The processor 12 maintains data structures in the memory subsystem 18 according to the application programs, the device driver programs, and operating system being executed.

15 The clock generator circuit 14 is coupled to a crystal device 30. The crystal device 30 enables generation of the processor clock 32 according to the contents of a clock speed register contained in the clock generator circuit 14. The processor 12 controls the frequency of  
20 the processor clock 32 by writing to the clock speed register in the clock generator circuit 14 over the system bus 28.

The crystal device 30 oscillates at a fixed predetermined frequency when electrical current is applied. The clock generator circuit  
25 14 adjusts the frequency of an oscillating signal generated by the crystal device 30 according to the contents of the clock speed register. For one embodiment, the processor 12 writes to the clock speed register to divide the frequency of the oscillating signal generated by the crystal device 30 by one, two, four, or eight. The processor 12 also  
30 writes to the clock speed register to stop the processor clock 32.

**Figure 2** is a block diagram that illustrates the software architecture of the computer system 10 for one embodiment. The software implemented on the computer system 10 includes an

operating system 48, and a set of application programs such as a pair of application programs 50 and 52 and associated device driver programs. The software architecture of the computer system 10 also includes a set of basic input/output software (BIOS) 40.

5

The operating system 48 and the application programs 50 and 52 and associated device drivers are loaded from the disk subsystem 20 and are executed from the memory subsystem 18. For one embodiment, the operating system 48 is Microsoft Windows.

10

The operating system 48 provides a set of system services to the application programs 50 and 52 including memory management and multi tasking functions. The operating system 48 enables the application programs 50 and 52 and associated device driver programs to access the input/output devices of the computer system 10 through the BIOS 40.

15

The BIOS 40 includes basic hardware driver routines for accessing and controlling the disk subsystem 20, the keyboard and mouse 22, and the display 24. The BIOS 40 also includes a clock generator driver 42. The clock generator driver 42 comprises a set of hardware driver routines for accessing a clock speed register 34 in the clock generator circuit 14.

20

The software architecture of the computer system 10 further comprises a performance manager 44, a performance control panel 54, and a performance state table 56. For one embodiment, the performance manager 44 is an extension of the operating system 48 that is loaded from the disk subsystem 20 and executed from the memory subsystem 18. The performance manager 44 processes requests from the application programs 50 and 52 to change the performance state for the processor 12.

25

30

For one embodiment, the application programs 50 and 52 execute a `set_performance_state` system call via the operating system 48 to request changes in the processor performance state by the performance manager 44. The `set_performance_state` system call to  
5 the performance manager 44 specifies either a high performance state, a medium performance state, or low performance state for the processor 12.

The performance manager 44 accesses the performance state  
10 table 56 to determine the processor clock 32 frequencies that correspond to the high, medium, and low performance states of the processor 12. The performance manager 44 then invokes the routines of the clock generator driver 42 to program the clock speed register 34 for the specified frequency of the processor clock 32.

15

For another embodiment, a cursor monitor 46 monitors a cursor generated on the display 24 by the operating system 48 or a cursor generated on the display 24 by one of the application programs 50 and 52. In this embodiment, the cursor monitor 46 monitors a `set_cursor`  
20 system call to the operating system 48 to determine the type of cursor displayed on the display 24. The cursor monitor 46 hooks to the `set_cursor` system call to detect changes to the cursor display by other application programs or driver programs executing on the processor 12.

25

A cursor generated on the display 24 in the shape of an hour glass indicates that the processor 12 is executing processor intensive software. The cursor monitor 46 generates a `set_performance_state` system call to the performance manager 44 to step up the performance state of the processor 12 to the high performance state if the hourglass  
30 cursor is being generated on the display 24. The cursor monitor 46 thereafter executes a `set_performance_state` system call to the performance manager 44 to restore the performance state of the processor 12 to the low performance state after the operating system 48

or the application programs 50 or 52 removes the hourglass cursor from the display 24.

**Figures 3a-3b** illustrate a control panel 70 generated on the display device 24 by the performance control panel 54. The performance control panel 54 is an application program that enables a user of the computer system 10 to select from a set of predetermined frequencies of the processor clock 32 that correspond to the high, medium, and low performance states of the processor 12. The performance control panel 54 employs the display and interactive input functions provided by the operating system 48 to generate the control panel 70 and to enter the processor clock frequencies.

For the example control panel 70 user selection shown in **Figure 3a**, the high performance state of the processor 12 corresponds to the processor clock 32 frequency of 25 MHz. The medium performance state corresponds to the processor clock 32 frequency of 20 MHz, and the low performance state corresponds to the processor clock 32 frequency of 15 MHz. The pulldown menu areas 71-73 of the control panel 70 enable the user to select from a set of four predetermined frequencies for the processor clock 32 as shown in **Figure 3b**.

For one embodiment, the predetermined frequencies available through the control panel 70 include the oscillation frequency of the crystal device 30,  $1/2$  the oscillation frequency of the crystal device 30,  $1/4$  of the oscillation frequency of the crystal device 30, and  $1/8$  the oscillation frequency of the crystal device 30. The control panel 70 enables the user to select any of the predetermined processor clock 32 frequencies for each of the high, medium, and low processor states.

The user employs a set of pulldown menu areas 71-73 of the control panel 70 to select a desired processor clock 32 frequency for each of the high, medium, and low performance states of the processor

12. If maximum performance for the processor 12 is always desired, then the user selects the fastest processor clock 32 frequency for the high, medium, and low processor states of the processor 12. On the other hand, if conservation of power is always desired, then the user  
5 selects the slowest processor clock 32 frequency for the high, medium, and low processor states of the processor 12. In addition, the user can select the processor clock 32 frequencies for the high, medium, and low processor states of the processor 12 to scale the balance between maximum performance and maximum power conservation for the  
10 computer system 10 as desired.

The performance control panel 54 transfers the clock speeds selected through the control panel 70 to the performance manager 44. The performance manager 44 then stores the selected clock  
15 frequencies for the high, medium, and low performance states in the performance state table 56. The performance manager 44 accesses the clock speeds from the performance state table 56 as set\_performance\_state system calls are received from the application programs 50 and 52 and the cursor monitor 46. The performance  
20 manger 44 then uses the clock generator driver 42 to write the appropriate value into the clock speed register 34 to set the selected processor clock 32 frequency.

For another embodiment, the clock speeds for the performance  
25 state table 56 are selected by a clock speed function in the BIOS 40 that enables user selections of clock speeds via command input.

The application programs 50 and 52 generate an information window on the display 24 while executing a processor intensive  
30 function. The information window enables the user to accelerate the processor clock 32 frequency during the processor intensive function.

**Figure 4** illustrates an information window 80 for one embodiment. The application program 50 is a word processing

application program that generates the information window 80 on the display 24 while performing a spell check on a document. The spell check function on a document is an example of a processor intensive function performed by the application program 50.

5

The information window 80 enables the user to accelerate the spell check function by selecting the accelerate button of the information window 80 using the keyboard and mouse subsystem 22. The information window 80 also enables the user to cancel the spell  
10 check function by selecting the cancel button of the information window 80 using the keyboard and mouse subsystem 22.

If the user selects the accelerate button of the information window 80, then the application program 50 executes a  
15 set\_performance\_state system call to the performance manager 44 to step up the performance state of the processor 12 to the high performance state. After completion of the processor intensive spell check function, the application program 50 executes a  
20 set\_performance\_state system call to the performance manager 44 to restore the processor 12 to the low or medium performance state of the processor 12.

**Figure 5** is a flow diagram that illustrates the handling of the processor 12 performance states by the application program 50 for one  
25 embodiment. At decision block 100, the application program 50 determines whether the user has initiated a processor intensive function of the application program 50. If a processor intensive function has not been selected at decision block 100, then the application program 50 continues execution at block 102. If a processor intensive  
30 function is selected at decision block 100, then control proceeds to block 104.

At block 104, the application program 50 executes a set\_performance\_state system call to the performance manager 44 to

step up the performance state of the processor 12 to the high performance state. Thereafter, the application program 50 performs the processor intensive function specified by the user.

5           At decision block 106, the application program 50 determines whether the processor intensive function is complete. If the processor intensive function completes at decision block 106, then control proceeds to block 108. At block 108, the application program 50 executes a `set_performance_state` system call to the performance  
10           manager 44 to restore the performance state of the processor 12 to the low performance state or the medium performance state as required by the application program 50. Thereafter, the application program 50 continues execution at block 110.

15           **Figure 6** is a flow diagram that illustrates the functions of the performance manager 44 for one embodiment. At block 140, the performance manager 44 initially reads the clock speed for the low performance state from the performance state table 56. Thereafter at  
20           block 142, the performance manager 44 invokes the clock generator driver 42 to write the specified clock speed for the low performance state to the clock speed register 34 of the clock generator circuit 14.

          At decision block 144, the performance manager 44 waits for a `set_performance_state` system call from the application programs 50  
25           and 52 and the cursor monitor 46 to change the performance state of the processor 12. If a `set_performance_state` system call to the performance manager 44 is received at decision block 144, then control proceeds to block 146.

30           At block 146 the performance manager 44 reads the clock speed for the performance state specified by the `set_performance_state` system call from the performance state table 56. Thereafter, at block 148, the performance manager 44 writes the specified clock speed from the performance state table 56 to the clock speed register 34 by



invoking the clock generator driver 42. Thereafter, the performance manager 44 returns to decision block 144 to wait for another set\_performance\_state system call.

- 5           In the foregoing specification the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention as set forth in the appended claims. The specification and  
10 drawings are accordingly to be regarded as illustrative rather than a restrictive sense.

**CLAIMS**

What is claimed is:

1. A computer system, comprising:
  - 5 clock generator circuit having a clock speed register and circuitry for generating a processor clock signal at a frequency determined by the clock speed register;  
processor coupled to receive the processor clock signal such that the processor clock signal synchronizes the processor, the
  - 10 processor coupled to fetch and execute a performance manager program that writes the clock speed register according to a performance state selected by an application program executing on the computer system.
- 15 2. The computer system of claim 1, wherein the processor writes the clock speed register with a value specified in a performance state table maintained by the performance manager program.
3. The computer system of claim 2, wherein the performance state
- 20 table stores a set of clock frequencies that correspond to a set of performance states for the processor.
4. The computer system of claim 3, further comprising a
- 25 performance control panel program for execution on the computer system that enables a user to select the clock frequencies for the performance states of the processor.
5. The computer system of claim 4, wherein the performance
- 30 manager program writes the clock speed register with one of the clock frequencies corresponding to a high performance state specified by a system call from the application program as the application program begins a processor intensive function.

6. The computer system of claim 4, wherein the performance manager program writes the clock speed register with one of the clock frequencies corresponding to a low performance state specified by a system call from the application program as the application program ends a processor intensive function.

7. The computer system of claim 4, wherein the performance manager program writes the clock speed register by invoking a clock generator driver program of a set of basic input/output software for the computer system.

8. The computer system of claim 4, wherein the application program generates an accelerate button on a display device, such that a user selects the accelerate button to cause the performance manager program to write the clock speed register with one of the clock frequencies corresponding to a high performance state of the processor.

9. A method for managing performance and power consumption of a computer system, comprising the steps of:

generating a processor clock signal at a frequency determined by a clock speed register;

synchronizing a processor of the computer system with the processor clock signal;

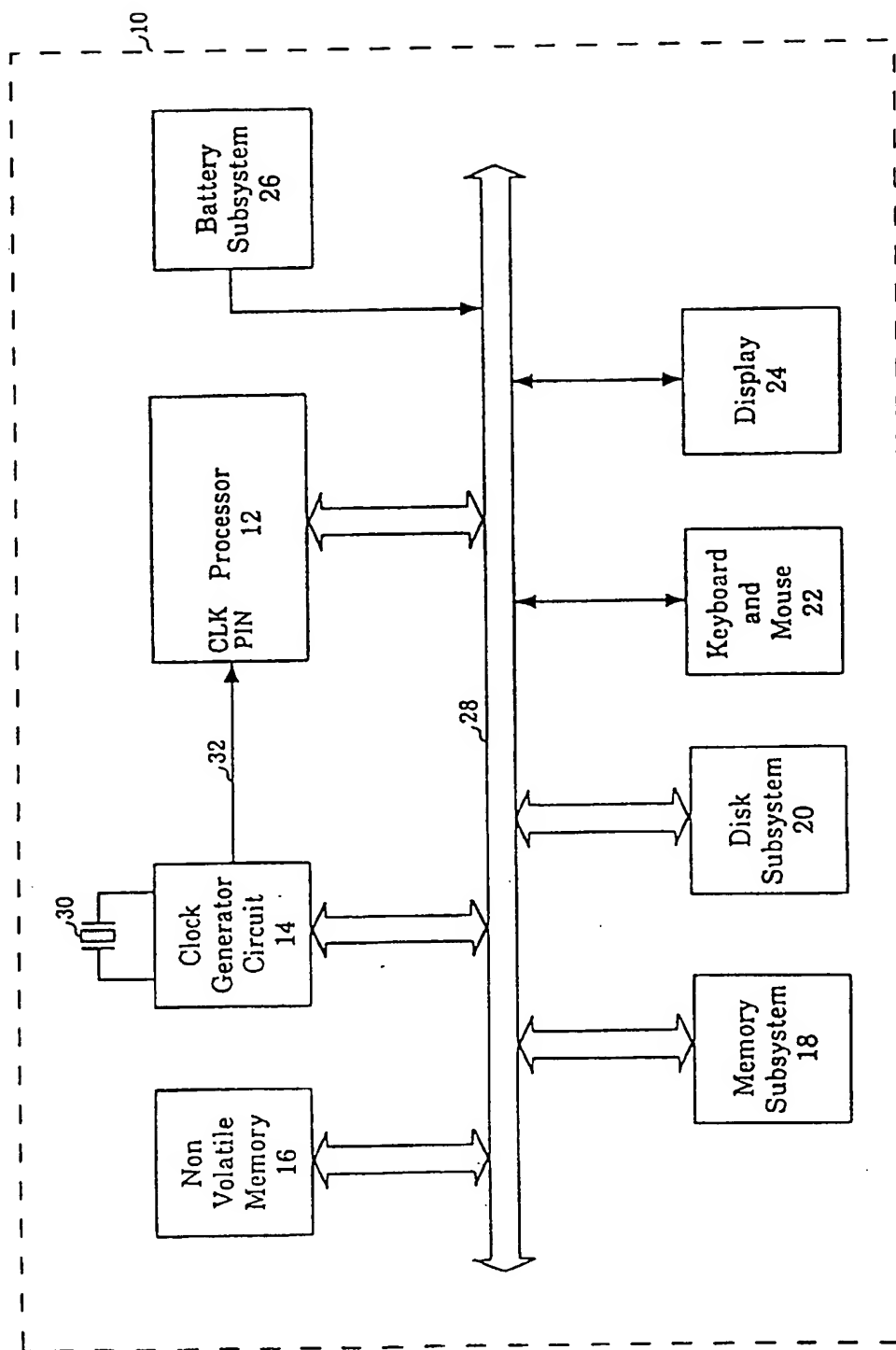
writing the clock speed register according to a performance state selected by an application program executing on the computer system.

10. The method of claim 9, wherein the step of writing the clock speed register according to a performance state selected by an application program comprises the step of writing the clock speed register with a value specified in a performance state table maintained by a performance manager program.

11. The method of claim 10, wherein the performance state table stores a set of clock frequencies that correspond to a set of performance states for the processor.
- 5 12. The method of claim 11, further comprising the step of selecting the clock frequencies for the performance states of the processor using a performance control panel program executed on the computer system.
- 10 13. The method of claim 12, wherein the step of writing the clock speed register according to a performance state selected by an application program comprises the step of writing the clock speed register with one of the clock frequencies corresponding to a high performance state specified by a system call from the application
- 15 program as the application program begins a processor intensive function.
14. The method of claim 12, wherein the step of writing the clock speed register according to a performance state selected by an
- 20 application program comprises the step of writing the clock speed register with one of the clock frequencies corresponding to a low performance state specified by a system call from the application program as the application program ends a processor intensive function.
- 25 15. The method of claim 12, wherein the step of writing the clock speed register according to a performance state selected by an application program comprises the step of invoking a clock generator driver program of a set of basic input/output software for the computer
- 30 system.
16. The method of claim 12, further comprising the step of generating an accelerate button on a display device, such that a user selects the accelerate button to cause the performance manager program to write

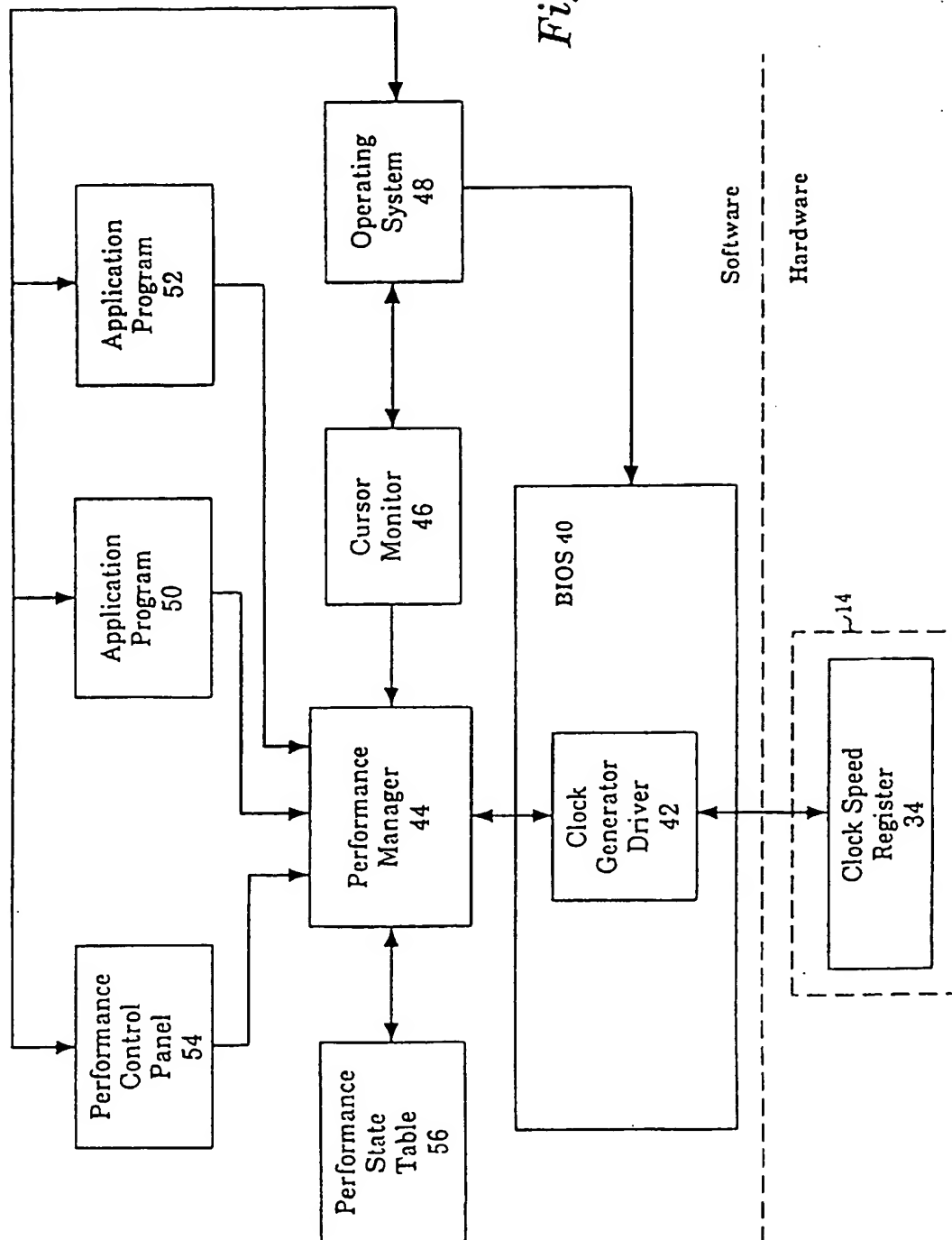
the clock speed register with one of the clock frequencies corresponding to a high performance state of the processor.

1 / 6

*Figure 1*

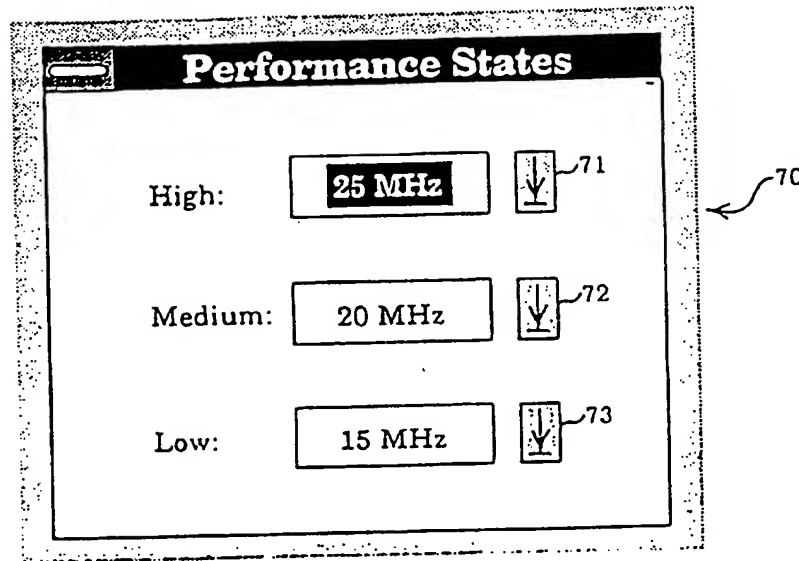
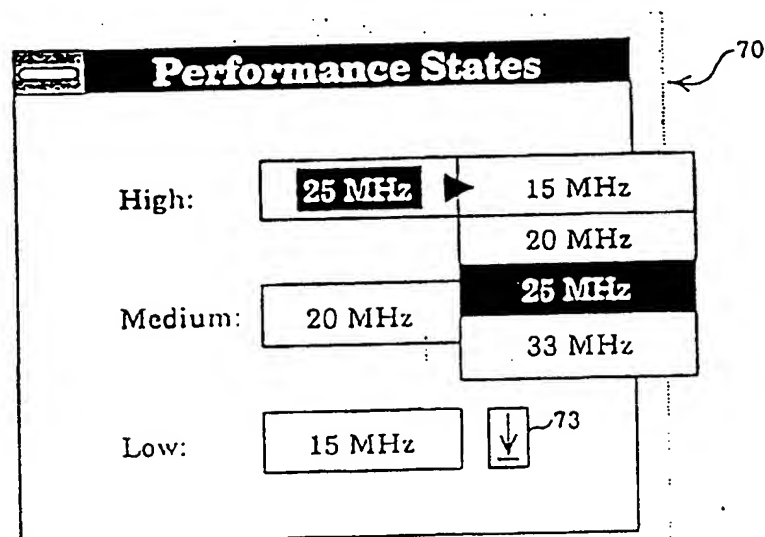
2 / 6

Figure 2



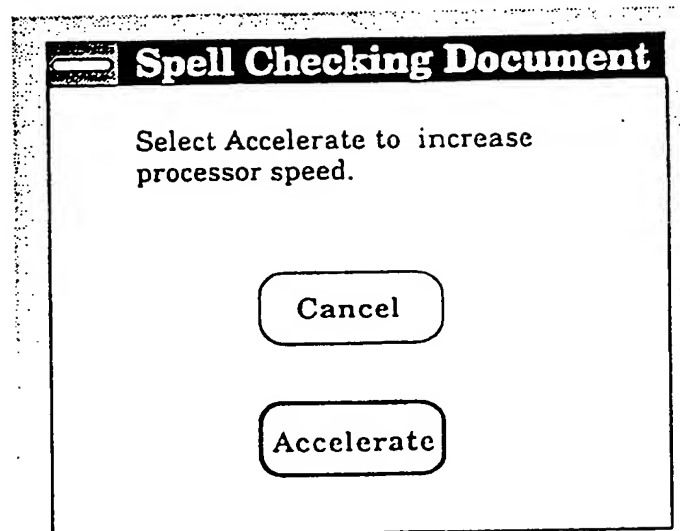
3 / 6

BEST AVAILABLE COPY

*Figure 3a**Figure 3b*

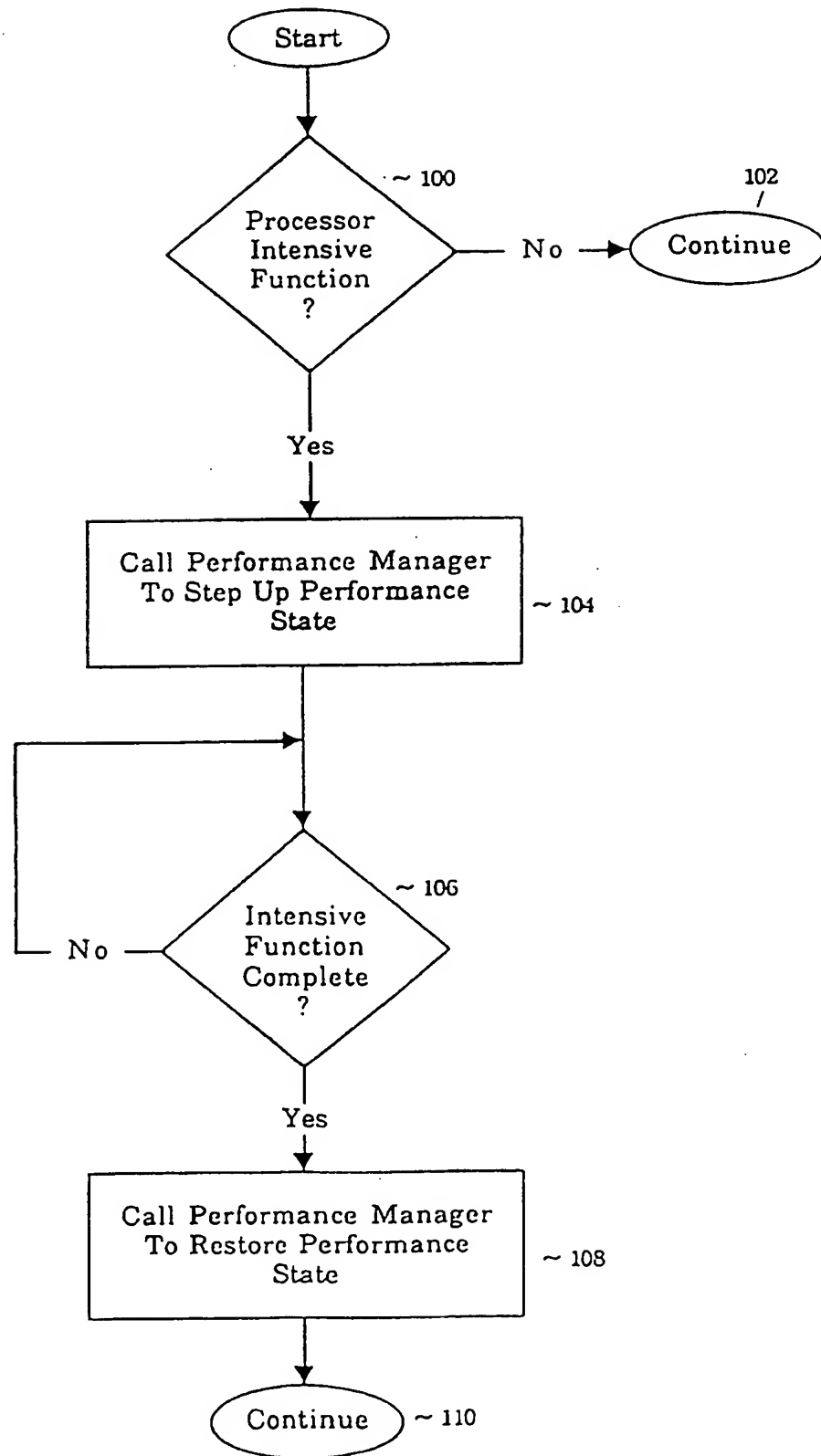


BEST AVAILABLE COPY

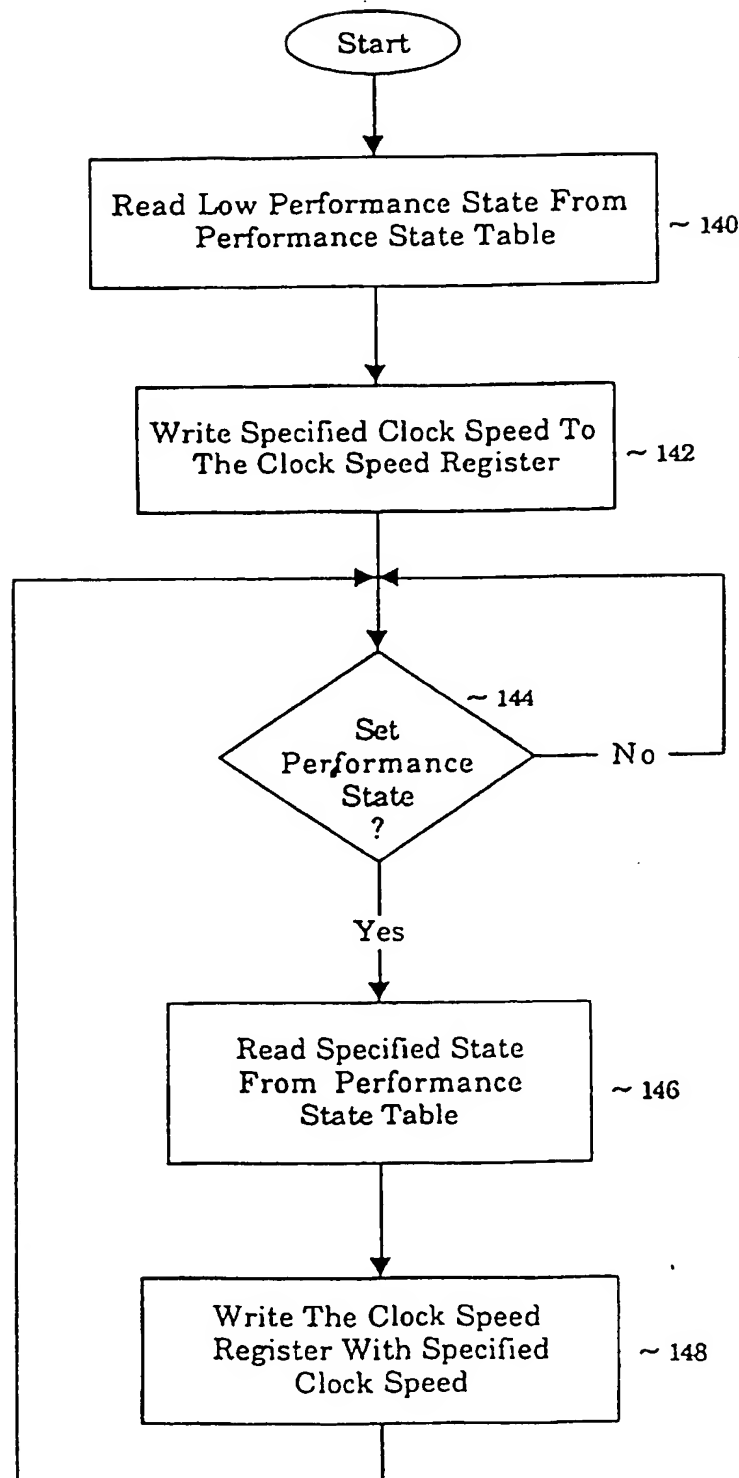


*Figure 4*

5 / 6

*Figure 5*

6 / 6

*Figure 6*

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/US95/07668

**A. CLASSIFICATION OF SUBJECT MATTER**

IPC(6) :G06F 13/00

US CL :395/750

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 395/750

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US, 5,086,387 (ARROYO ET AL) 04 February 1992, abstract, col. 2 lines 5-8.	1-16
Y	US, 5,142,684 (PERRY ET AL) 25 August 1992, fig 2c.	1-16
Y	US, 5,230,074 (CANOVA, JR. ET AL) 20 July 1993, col. 7 lines 9-25 and 52-54, figs 4-5.	2-8,10-16
Y	US, 5,292,341 (SNELL) 08 March 1994, col. 17 line 65 through col. 18 line 3, fig 9.	4-8, 12-16
Y	US, 4,775,935 (YOURICK) 04 October 1988, col. 13 lines 26-27.	8, 16



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:	*T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
*A* document defining the general state of the art which is not considered to be part of particular relevance	*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
*E* earlier document published on or after the international filing date	*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
*L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	*A* document member of the same patent family
*O* document referring to an oral disclosure, use, exhibition or other means	
*P* document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

24 AUGUST 1995

Date of mailing of the international search report

20 SEP 1995

Name and mailing address of the ISA/US  
Commissioner of Patents and Trademarks  
Box PCT  
Washington, D.C. 20231

Facsimile No. (703) 308-5358

Authorized officer

JOHN TRAVIS

Telephone No. (703) 308-5212